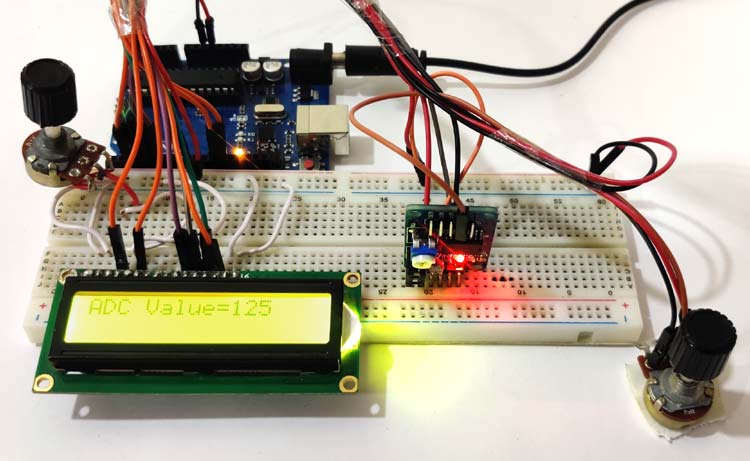
**Подключение АЦП/ЦАП модуля PCF8591 к Arduino**

Опубликовано [20.12.2020](https://microkontroller.ru/arduino-projects/podklyuchenie-aczp-czap-modulya-pcf8591-k-arduino/) автором [admin-new](https://microkontroller.ru/author/admin-new/" \o "Посмотреть все записи автора admin-new)

Аналого-цифровое преобразование играет исключительно важную роль в современных электронных устройствах. К примеру, многие датчики выдают на свой выход аналоговый сигнал, а микроконтроллеры могут оперировать только с цифровыми (двоичными) значениями, поэтому чтобы согласовать выход такого датчика со входом микроконтроллера, необходимо использование аналого-цифрового преобразователя (АЦП).

Некоторые микроконтроллеры, например, AVR, Arduino, MSP430, PIC16F877A, имеют в своем составе встроенные АЦП. Однако некоторые современные микроконтроллеры не имеют встроенных АЦП, например, 8051, Raspberry Pi. Поэтому с такими микроконтроллерами используют внешние АЦП, например, микросхемы ADC0804, ADC0808.



На нашем сайте вы можете посмотреть следующие проекты, использующие АЦП/ЦАП:

* [использование аналого-цифрового преобразования (АЦП) в Arduino Uno](https://microkontroller.ru/arduino-projects/ispolzovanie-analogo-czifrovogo-preobrazovaniya-v-arduino-uno/);
* [использование аналого-цифрового преобразователя (АЦП) в микроконтроллере AVR ATmega16](https://microkontroller.ru/avr-microcontroller-projects/kak-ispolzovat-adc-v-avr-atmega16/);
* [подключение цифро-аналогового преобразователя (ЦАП) MCP4725 к Arduino](https://microkontroller.ru/arduino-projects/podklyuchenie-czifro-analogovogo-preobrazovatelya-czap-mcp4725-k-arduino/).

В этой же статье мы рассмотрим подключение АЦП/ЦАП модуля PCF8591 к плате Arduino.

**Необходимые компоненты**

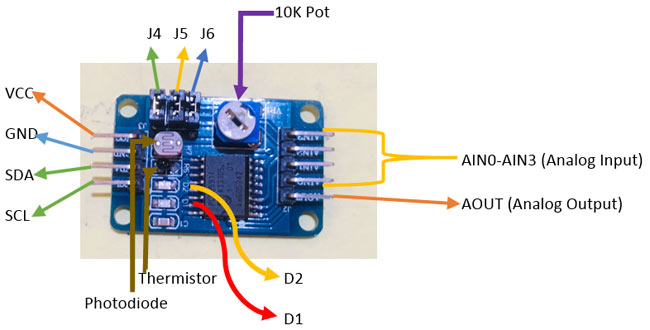
1. Плата Arduino Uno ([купить на AliExpress](http://alii.pub/5uhymx)).
2. АЦП/ЦАП модуль PCF8591 (PCF8591 ADC/DAC Module) ([купить на AliExpress](http://alii.pub/5uw281)).
3. Потенциометр 100 кОм – 2 шт. ([купить на AliExpress](http://ali.pub/5cb5ut)).
4. Соединительные провода.

**АЦП/ЦАП модуль PCF8591**

PCF8591 представляет собой 8-битный аналого-цифровой (АЦП) или цифро-аналоговый преобразователь (ЦАП), соответственно, диапазон возможных значений после АЦП составляет для него от 0 до 255. Также модуль содержит фоторезистор и терморезистор (термистор). Модуль имеет 4 аналоговых входа и один аналоговый выход. Модуль работает по [протоколу I2C](https://microkontroller.ru/arduino-projects/ispolzovanie-interfejsa-i2c-v-arduino-polnoe-rukovodstvo/), соответственно, линия SCL используется для синхронизации, а линия SDA – для передачи данных. Модуль работает от питающего напряжения 2.5-6V и имеем низкое энергопотребления в режиме ожидания (бездействия).

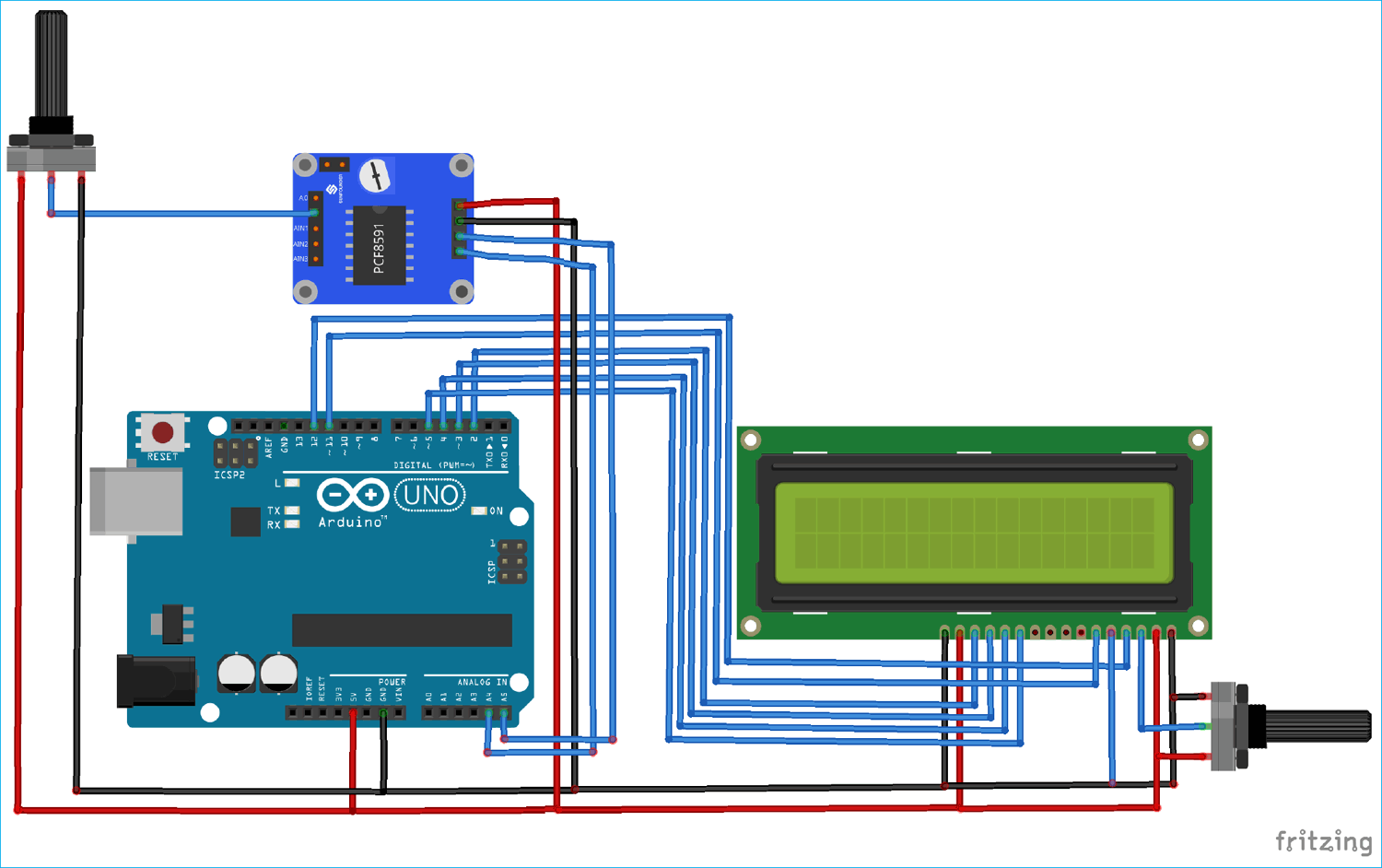
Значением входного напряжения можно управлять с помощью ручки потенциометра, присутствующего в составе модуля. Также в составе модуля есть три переключателя: J4 – позволяет подключать цепь термистора, J5 – позволяет подключать цепь фоторезистора, J6 – позволяет подключать цепь регулировки напряжения. Для доступа к этим цепям можно использовать адреса этих переключателей (джамперов): 0x50 для J6, 0x60 для J5 и 0x70 для J4. В составе модуля есть два светодиода: D1 показывает интенсивность (уровень) выходного напряжения, а D2 показывает уровень питающего напряжения. То есть чем больше уровни выходного и питающего напряжений, тем, соответственно, более ярко горят светодиоды D1 и D2.

Внешний вид и назначение контактов (распиновка) модуля PCF8591 показаны на следующем рисунке.



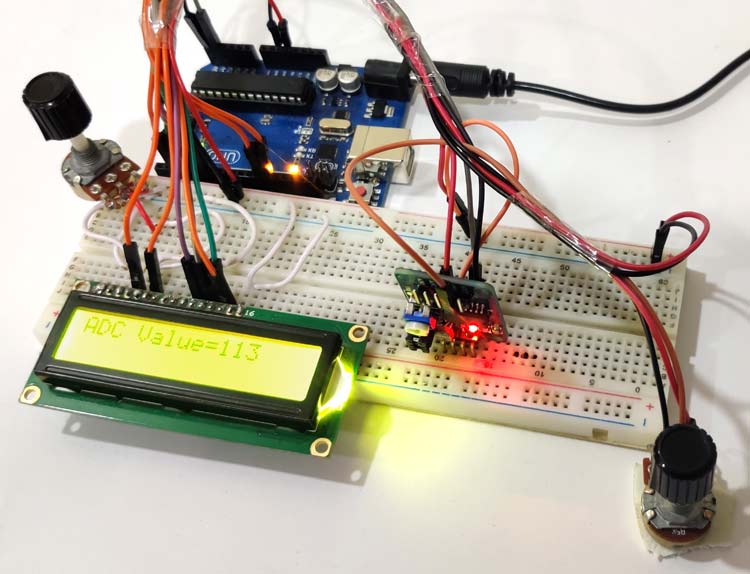
**Схема проекта**

Схема подключения АЦП/ЦАП модуля PCF8591 к плате Arduino представлена на следующем рисунке.

[](https://microkontroller.ru/wp-content/uploads/2020/12/circuit-diagram-for-interfacing-pcf8591-adc-with-arduino.png)

Как видите, схема достаточно проста. В этой схеме мы будем считывать аналоговые значения с любого из аналоговых контактов и изменять эти значения с помощью потенциометра 100 кОм. Контакты VCC и GND модуля подключены к контактам 5V и GND платы Arduino. Контакты SDA и SCL модуля подключены к контактам A4 и A5 платы Arduino – это обеспечивает связь по протоколу I2C. Средний контакт потенциометра подключен к контакту AIN0 модуля PCF8591. Контакты данных (D4-D7) ЖК дисплея подключены к цифровым контактам D5-D2 платы Arduino, а контакты RS и EN ЖК дисплея подключены к контактам D12 и D11 платы Arduino. Контакт V0 ЖК дисплея подключен к потенциометру, который используется для регулировки его яркости.

Внешний вид собранной конструкции проекта показан на следующем рисунке.



**Объяснение программы для Arduino**

Полный код программы и видео, демонстрирующее работу схемы, приведены в конце статьи, здесь же мы кратко рассмотрим основные фрагменты кода программы.

Первым делом в программе нам необходимо подключить библиотеки для использования возможностей протокола I2C и для работы с ЖК дисплеем.

|  |  |
| --- | --- |
| 1  2 | #include<Wire.h>  #include <LiquidCrystal.h> |

После этого определим (объявим) ряд макросов. В первом макросе будет записан адрес шины (адрес ведомого – в нашем случае это модуль PCF8591) для протокола I2C, а во втором – адрес первого входного контакта (AIN0) модуля PCF8591 (он подключен к потенциометру в схеме).

|  |  |
| --- | --- |
| 1  2 | #define PCF8591 (0x90 >> 1)  #define AIn0 0x00 |

Далее сообщим плате Arduino, к каким ее контактам подключен ЖК дисплей и инициализируем переменную, в котором будем хранить значение, считываемое с аналогового контакта модуля PCF8591.

|  |  |
| --- | --- |
| 1  2  3 | const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;  LiquidCrystal lcd(rs, en, d4, d5, d6, d7);  int Value = 0; |

В функции void setup() мы инициализируем связь по протоколу I2C и ЖК дисплей для работы в режиме 16х2.

|  |  |
| --- | --- |
| 1  2  3  4  5 | void **setup**()  {  Wire.begin();  lcd.begin(16,2);  } |

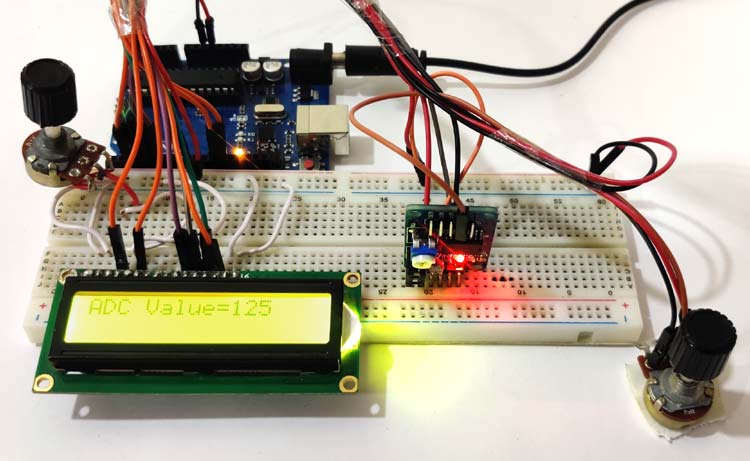
В функции void loop() мы начнем обмен данными с модулем PCF8591 с помощью команды Wire.beginTransmission. Далее мы дадим команду модулю PCF8591 на считывание аналоговых значений на своем входе AIN0 (вторая строка в коде). Затем мы завершим передачу с помощью функции  Wire.endTransmission и запросим измеренное аналоговое значение (Wire.requestFrom) с модуля PCF8591.

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | void **loop**()  {  Wire.beginTransmission(PCF8591);  Wire.write(AIn0);  Wire.endTransmission();  Wire.requestFrom(PCF8591, 1); |

Далее мы сохраним принятое с модуля PCF8591 значение в переменной Value и выведем это значение на экран ЖК дисплея.

|  |  |
| --- | --- |
| 1  2  3  4  5 | Value=Wire.read();  lcd.print("ADC Value=");  lcd.print(Value);  **delay**(500);  lcd.clear();} |

После подачи питания на схему и загрузки программы в плату Arduino вы сможете наблюдать считываемое аналоговое значение на экране ЖК дисплея. Изменять это значение можно с помощью ручки потенциометра.



**Исходный код программы (скетча)**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | #include <LiquidCrystal.h>  #include<Wire.h>  #define PCF8591 (0x90 >> 1)  #define AIN0 0x00  const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;  LiquidCrystal lcd(rs, en, d4, d5, d6, d7);  int Value = 0;  void **setup**()  {    Wire.begin();    lcd.begin(16,2);  }  void **loop**()  {    Wire.beginTransmission(PCF8591);    Wire.write(AIN0);    Wire.endTransmission();    Wire.requestFrom(PCF8591, 1);    Value = Wire.read();    lcd.print("ADC Value=");    lcd.print(Value);  **delay**(500);    lcd.clear();  } |